

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis system

Sistem harus mudah di akses, paradigma tidak bisa dengan query. Karena pengetahuan pengguna untuk menyusun query masih rendah. Oleh karena itu digunakan paradigma IR, dimana lebih mudah dalam menyusun query.

3.1.1. Kebutuhan User

Untuk analisis kebutuhan user ini agar aplikasi ini dapat membantu user untuk memperoleh informasi terkait obat-obat berbagai macam penyakit khususnya balita.

3.1.2. Kebutuhan Sistem

Analisis kebutuhan sistem sangat diperlukan dalam mendukung kinerja aplikasi, apakah aplikasi yang dibuat telah sesuai dengan kebutuhan atau belum. Karena kebutuhan sistem akan mendukung tercapainya tujuan suatu aplikasi.

Adapun kebutuhan sistem meliputi :

1. Webserver : merupakan komputer yang dikhususkan untuk menaruh data obat-obatan. Semua komputernya logikanya bisa dijadikan server, namun server yang khusus untuk menaruhdata obat-obatan ini punya spesifikasi khusus. Hal ini untuk menjaga kecepatan server stabil dan menaruh kriteria yang diharapkan.
2. Teknologi : Pengolahan / pemrosesan data obat-obatan dapat berupa mengkonversi (mengubah data ke bentuk lain), menganalisis (analisa kondisi), menghitung (kalkulasi), mensintesis (penggabungan) segala bentuk data dan informasi tentang obat-obatan penyakit balita.
3. Web platform : merupakan salah satu hal yang sangat penting dalam mendesign suatu website. Dalam mendesign suatu website ini harus ditentukan terlebih dahulu apa yang ingin kita tulis dalam website ini dan seperti apa design aplikasi pencarian obat obatan penyakit balita secara garis besar. Jadi isi dari website ini akan selalu sesuai dengan tema yang kita tentukan pertama kali.

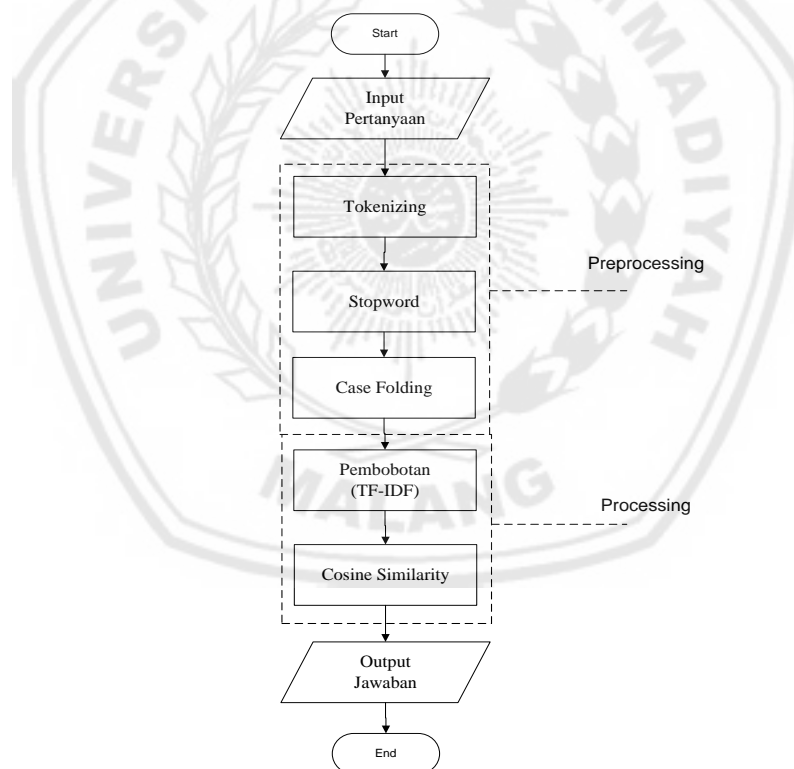
3.2 Perancangan Program Aplikasi

Program aplikasi dirancang untuk memperkenalkan tahapan-tahapan dalam Algoritma *Cosine Similarity* hingga didapatkannya hasil pencarian yang relevan. *User* diminta untuk menginput *query* ke dalam kotak pencarian dan menekan tombol cari.

3.2.1 Data Yang Digunakan

Data yang digunakan adalah data Obat balita Umum yang akan dimasukkan dalam program aplikasi yang dirancang dengan pertimbangan bahwa analisa lebih mudah dilakukan pada data yang jumlahnya belum terlalu besar. Pencocokan hasil akan dilakukan pada setiap tahap algoritma *Cosine Similarity* antara program aplikasi dengan perhitungan manual pada bab berikutnya.

3.2.2 Gambaran Sistem



Gambar 3.1 Flowchart Basis Pengetahuan

3.2.3 Diagram Alur Tahapan Algoritma *Cosine Similarity*

Berikut tahapan alur algoritma *Cosine Similarity*.

- 1) Dimulai dari menginputkan pertanyaan tentang gejala penyakit balita.

- 2) Setelah menginputkan pertanyaan selanjutnya yaitu proses tokenazing yang dimana proses pemotongan data input berupa string menjadi kata – kata tunggal berdasarkan spasi.
- 3) Selanjutnya dari pembacaan kata yang terdapat pada input pertanyaan untuk dilakukan proses *Stop Words Removal* dan *Case Folding*. Ini semua merupakan sub bagian dari tahap *Preprocessing*. Untuk menghapus kata tidak penting dan menjadikan input pertanyaan menjadi huruf kecil.
- 4) Setelah itu proses pembobotan TF-IDF yang membandingkan *term* hasil kali titik dari dua baris vector X mencerminkan bahwa dua *term* memiliki pola yang sama dalam sebuah dokumen pencarian obat.
- 5) Membandingkan dokumen pencarian obat: Hasil kali titik dari dua kolom vektor X.
- 6) Membandingkan term pencarian obat dengan dokumen indikasi obat.
- 7) Selanjutnya penghitungan skor terhadap dokumen pencarian obat dengan indikasi obat. Yang dimana hasil paling tinggi merupakan hasil yang sudah mencapai kemiripan dengan pencarian obat.

Berdasarkan gambar 3.1 tiap-tiap entitas *flowchart* diatas dapat di jelaskan seperti berikut:

a. *PREPROCESSING*:

- 1) *Tokenizing*: Proses pemotongan data input berupa string menjadi kata – kata tunggal berdasarkan spasi.
- 2) *Stopword Removal*: Berfungsi untuk menghapus kata tidak penting contoh (‘dan, yang, karena dan lain sebagainya’).
- 3) *Case folding*: Berfungsi untuk mengubah semua huruf dalam dokumen menjadi huruf kecil.
- 4) Membangun database TF-IDF
- 5) *Proses Cosine Similarity*: Berfungsi untuk mengetahui bobot dari term pada setiap data, bobot term tersebut di gunakan data proses perhitungan jarak kemiripannya.

b. *PROCESS*

Untuk penyelesaian Masalah Menggunakan *Cosine Similarity* meliputi:

Pencarian Obat Balita menggunakan metode *Cosine Similarity*, yang kemudian di uji dengan akurasi, presisi dan recall. Nilai hasil uji tersebut mengindikasikan bagaimana akurasi metode *Cosine Similarity* dalam menghitung antara kemiripan queri dengan Obat Balita atau biasa disebut dengan nilai evaluasi. Pada studi kasus ini, digunakan data Obat Balita yang sederhana untuk mempermudah perhitungan manual. Data yang digunakan ditunjukkan pada tabel 3.1.

Tabel 3.1 Tabel Contoh Obat Balita

No	Indikasi
1	Sanmol diindikasikan dapat meringankan rasa sakit di keadaan sakit kepala, sakit gigi serta menurunkan demam.
2	Menghilangkan gejala flu yang berhubungan dengan alergi sal nafas atas.
3	melebarkan saluran napas(bronkus) dan membantu mengeluarkan dahak pada kasus asma atau batuk karena asma terutama pada anak-anak

Berdasarkan data dicontohkan pencarian dengan keyword “menghilangkan gejala flu”

1. Tokenizing

Tahap ini merupakan tahap pertama pada *preprocessing*, proses *Tokenizing* merupakan tahap pemotongan string input untuk memisah kalimat menjadi kata dan penghapusan karakter tanda baca, berikut merupakan asumsi dari proses tokenizing yang di ambil dari salah satu data studi kasus yang terdapat pada table 3.1.

Q : query / pencarian obat dengan keyword gejala.

D1 : Indikasi obat pada dokumen 1.

D2 : Indikasi obat pada dokumen 2.

D3 : Indikasi obat pada dokumen 3.

Tabel 3.2 Hasil Tokenizing

Q	D1	D2	D3
Menghilangkan	Sanmol	Menghilangkan	Melebarkan
Gejala	diindikasikan	gejala	Saluran
Flu	Dapat	flu	napasbronkus
	meringankan	yang	Dan
	Rasa	berhubungan	Membantu
	Sakit	dengan	mengeluarkan
	Di	alergi	Dahak
	Keadaan	sal	Pada
	Sakit	nafas	Kasus
	kepala,	atas.	Asma
	Sakit		Atau
	Gigi		Batuk
	Serta		Karena
	menurunkan		Asma
	demam.		Terutama
			Pada
			Anak-anak

2. Stopword Removal

Tahap *stopword removal* merupakan tahap ke dua dari *preprocessing*. Dalam proses ini dilakukan proses menghilangkan kata yang tidak penting dalam teks. Untuk proses ini diperlukan suatu kamus kata-kata yang menyimpan kata-kata yang bisa dihilangkan. Berikut merupakan daftar kata tidak penting yang di sajikan dalam bentuk 3, dan hasil dari proses *stopword removal* akan disajikan pada Tabel 3.3.

Tabel 3.3 Daftar Kata Tidak Penting

No	Kata Tidak Penting
1	Di
2	Yang
3	Tidak

4	Untuk
5	Ini
6	Itu
7	Dari
8	dan
9	ke
10	atau
11	tak
12	akan
13	pada
14	dengan

Tabel 3.4 Hasil Stopword Removal

Q	D1	D2	D3
Menghilangkan	Sanmol	Menghilangkan	Melebarkan
Gejala	Diindikasikan	gejala	Saluran
Flu	Dapat	flu	napas(bronkus)
	Meringankan	berhubungan	Membantu
	Rasa	alergi	Mengeluarkan
	Sakit	sal	Dahak
	Keadaan	nafas	Kasus
	Sakit	atas.	Asma
	kepala,		Batuk
	Sakit		Karena
	Gigi		Asma
	Serta		Terutama
	Menurunkan		Anak-anak
	demam.		

3. Case folding

Tahap *Case Folding* merupakan tahap ketiga dari *preprocessing*, Proses Case Folding merupakan proses yang berfungsi untuk mengubah semua huruf dalam dokumen menjadi huruf kecil proses case folding akan di jelaskan dalam gambar alur seperti pada tabel 3.5 berikut:

Tabel 3.5 Hasil Case Folding

Q	D1	D2	D3
menghilangkan	Sanmol	menghilangkan	Melebarkan
Gejala	Diindikasikan	gejala	Saluran
Flu	Dapat	flu	napas(bronkus)
	Meringankan	berhubungan	Membantu
	Rasa	alergi	Mengeluarkan
	Sakit	sal	Dahak
	Keaadaan	nafas	Kasus
	Sakit	atas.	Asma
	kepala,		Batuk
	Sakit		Karena
	Gigi		Asma
	Serta		Utama
	Menurunkan		anak-anak
	demam.		

4. Proses Pembobotan (TF-IDF)

Dalam perhitungan bobot menggunakan TF-IDF, dihitung terlebih dahulu nilai TF perkata dengan bobot masing-masing kata adalah 1. Sedangkan nilai IDF diformulasikan pada persamaan 3.1. dimana $IDF(word)$ adalah nilai IDF dari setiap kata yang akan di cari, td adalah jumlah keseluruhan dokumen yang ada, df jumlah kemuculan kata pada semua dokumen. Setelah mendapat nilai TF dan IDF, maka untuk mendapatkan bobot akhir dari TF-IDF diformulasikan pada persamaan 3.2. dimana $w(word_i)$ adalah nilai bobot dari setiap kata, $TF(word_i)$ adalah hasil perhitungan dari TF. IDF_i adalah hasil dari perhitungan IDF. Berikut merupakan proses perhitung bobot menggunakan metode tf-idf:

Tahap pertama dari proses perhitungan bobot menggunakan tf-idf adalah menghitung IDF, persamaan yang digunakan untuk menghitung IDF adalah persamaan 3.1. table 3.6 berikut merupakan hasil hitung IDF:

Tabel 3.6 Hasil Hitung IDF

	Tf					Idf
Term	Q	D1	D2	D3	df	Log(N/df)
Hilang	1		1		1	$\text{Log}(4/1) = 0.602059991$
Gejala	1		1		1	$\text{Log}(4/1) = 0.602059991$
Flu	1		1		1	$\text{Log}(4/1) = 0.602059991$
Sanmol		1			1	$\text{Log}(4/1) = 0.602059991$
Indikasi		1			1	$\text{Log}(4/1) = 0.602059991$
Dapat		1			1	$\text{Log}(4/1) = 0.602059991$
Ringan		1			1	$\text{Log}(4/1) = 0.602059991$
Rasa		1			1	$\text{Log}(4/1) = 0.602059991$
Ada		1			1	$\text{Log}(4/1) = 0.602059991$
Sakit		3			3	$\text{Log}(4/3) = 0.124938737$
Kepala		1			1	$\text{Log}(4/1) = 0.602059991$
Gigi		1			1	$\text{Log}(4/1) = 0.602059991$
Serta		1			1	$\text{Log}(4/1) = 0.602059991$
Turun		1			1	$\text{Log}(4/1) = 0.602059991$
demam.		1			1	$\text{Log}(4/1) = 0.602059991$
Hubung			1		1	$\text{Log}(4/1) = 0.602059991$
Alergi			1		1	$\text{Log}(4/1) = 0.602059991$
Sal			1		1	$\text{Log}(4/1) = 0.602059991$
Nafas			1		1	$\text{Log}(4/1) = 0.602059991$
atas.			1		1	$\text{Log}(4/1) = 0.602059991$
Lebar				1	1	$\text{Log}(4/1) = 0.602059991$
Salur				1	1	$\text{Log}(4/1) = 0.602059991$
napas(bronkus)				1	1	$\text{Log}(4/1) = 0.602059991$
Bantu				1	1	$\text{Log}(4/1) = 0.602059991$
Keluar				1	1	$\text{Log}(4/1) = 0.602059991$
Dahak				1	1	$\text{Log}(4/1) = 0.602059991$
Kasus				1	1	$\text{Log}(4/1) = 0.602059991$
Asma				2	2	$\text{Log}(4/2) = 0.301029996$
Ata				1	1	$\text{Log}(4/1) = 0.602059991$
Batuk				1	1	$\text{Log}(4/1) = 0.602059991$
Karena				1	1	$\text{Log}(4/1) = 0.602059991$

Utama				1	1	$\text{Log}(4/1) = 0.602059991$
Anak-anak				1	1	$\text{Log}(4/1) = 0.602059991$

Tahap kedua dari proses perhitung bobot tf-idf adalah tahap perkalian antara tf (term frekuensi) atau jumlah kemunculan kata tiap dokumen dikali dengan idf, persamaan yang di gunakan dalam proses ini dapat di lihat pada persamaan 2.2. Pada table 3.6 Berikut merupakan hasil dari perhitungan pembobotan kata tf-idf:

Tabel 3.7 Hasil TF*IDF

Term	Q	D1	D2	D3
Hilang	0.60206	0	0.60206	0
Gejala	0.60206	0	0.60206	0
Flu	0.60206	0	0.60206	0
Sanmol	0	0.60206	0	0
Indikasi	0	0.60206	0	0
Dapat	0	0.60206	0	0
Ringan	0	0.60206	0	0
Rasa	0	0.60206	0	0
Ada	0	0.60206	0	0
Sakit	0	0.374816	0	0
Kepala	0	0.60206	0	0
Gigi	0	0.60206	0	0
Serta	0	0.60206	0	0
Turun	0	0.60206	0	0
demam.	0	0.60206	0	0
Hubung	0	0	0.60206	0
Alergi	0	0	0.60206	0
Sal	0	0	0.60206	0
Nafas	0	0	0.60206	0
atas.	0	0	0.60206	0
Lebar	0	0	0	0.60206
Salur	0	0	0	0.60206
napas(bronkus)	0	0	0	0.60206
Bantu	0	0	0	0.60206

Keluar	0	0	0	0.60206
Dahak	0	0	0	0.60206
Kasus	0	0	0	0.60206
Asma	0	0	0	0.60206
Ata	0	0	0	0.60206
Batuk	0	0	0	0.60206
Karena	0	0	0	0.60206
Utama	0	0	0	0.60206
Anak-anak	0	0	0	0.60206

5. Proses Cosine Similarity

Dalam proses Cosine Similarity yang menjadi masukan adalah bobot dari term setiap data, bobot term tersebut di gunakan dala proses perhitungan jarak kemiripan dengan kata cluster, kemudian dari setiap nilai akan menentukan centroid setiap cluster. Tahap pertama dalam proses Cosine Similarity adalah menghitung jarak panjang vector dengan cara mengalikan bobot tf-idf query dengan bobo tf-idf setiap dokumen, table 3.8 berikut merupakan hasil perhitungan panjang vector :

Tabel 3.8 Hasil Hitung Panjang Vektor

Term	Q	D1	D2	D3
hilang	0.362476	0	0.362476233	0
Gejala	0.362476	0	0.362476233	0
flu	0.362476	0	0.362476233	0
Sanmol	0	0	0	0
indikasi	0	0	0	0
dapat	0	0	0	0
ringan	0	0	0	0
rasa	0	0	0	0
ada	0	0	0	0
sakit	0	0	0	0
kepala	0	0	0	0
gigi	0	0	0	0
serta	0	0	0	0

turun	0	0	0	0
demam.	0	0	0	0
hubung	0	0	0	0
alergi	0	0	0	0
sal	0	0	0	0
nafas	0	0	0	0
atas.	0	0	0	0
lebar	0	0	0	0
salur	0	0	0	0
napas(bronkus)	0	0	0	0
bantu	0	0	0	0
keluar	0	0	0	0
dahak	0	0	0	0
kasus	0	0	0	0
asma	0	0	0	0
ata	0	0	0	0
batuk	0	0	0	0
karena	0	0	0	0
utama	0	0	0	0
Anak-anak	0	0	0	0
Jumlah panjang vektor	1.087429	0	1.087428699	0
Hasil akar dari vektor	1.042798	0	1.042798494	0

Panjang vector di hitung berdasarkan rumus seperti pada proses berikut ini:

$$\text{Panjang vektor Q} = \sqrt{1.0874} = 1.0427$$

$$\text{Panjang vektor D1} = \sqrt{0} = 0$$

$$\text{Panjang vektor D2} = \sqrt{1.0874} = 1.0427$$

$$\text{Panjang vektor D3} = \sqrt{0} = 0$$

Tahap kedua dari proses Cosine Similarity adalah

menghitung jarak kemiripan. Table 3.6, table 3.7, dan table 3.8 Berikut merupakan hasil hitung jarak setiap kelas terhadap dokumen.

Hasil Jarak keyword terhadap dokumen di dapat :

$$\begin{aligned}\text{Cos (Q,D1)} &= \frac{1.0874}{\sqrt{1.0874 \times 0}} = \frac{1.0874}{1.0427 \times 0} = 0 \\ \text{Cos (Q,D2)} &= \frac{1.0874}{\sqrt{1.0874 \times 1.0874}} = \frac{1.0874}{1.0427 \times 1.0427} = 1 \\ \text{Cos (Q,D3)} &= \frac{1.0874}{\sqrt{1.0874 \times 0}} = \frac{1.0874}{1.0427 \times 0} = 0\end{aligned}$$

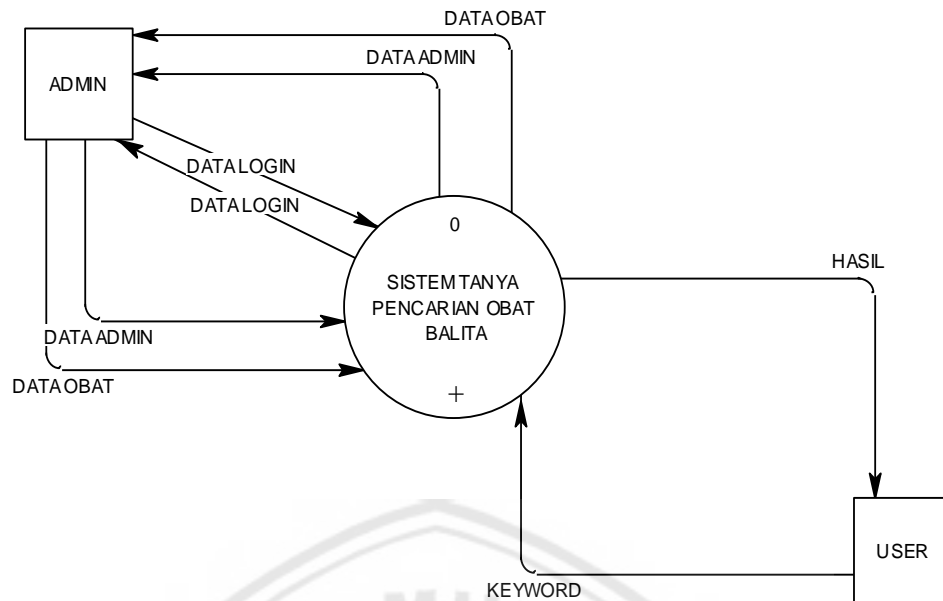
Berdasarkan hasil hitung kemiripan diatas disimpulkan bahwa yang memiliki nilai tertinggi merupakan hasil kemiripan terbaik sehingga D2 merupakan hasil yang paling mirip dengan kata kunci yang di gunakan.

- 1) *Cosine Similarity*: Berfungsi untuk menghitung kemiripan antara dokumen yang di cari.
- 2) Hasil Cari: Berfungsi untuk menampilkan dan mengirim sms hasil sesuai dengan pencarian algoritma *Cosine Similarity*.

3.2.4 Data Flow Diagram (DFD) Aplikasi Administrator menggunakan Web

a. Data Flow Diagram Level 0 (Diagram Konteks)

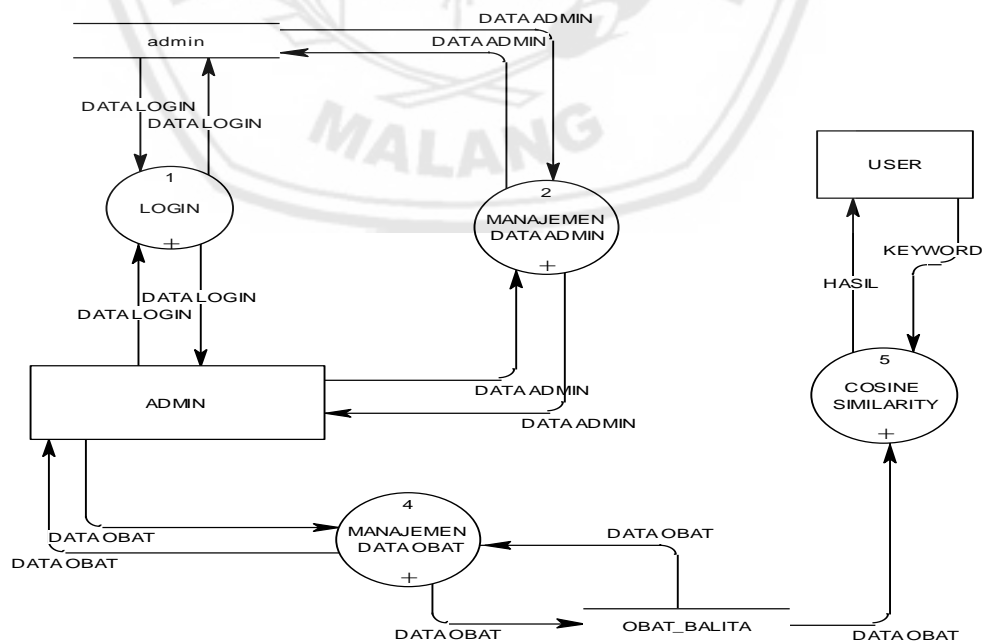
Pada DFD level 0 ini terdapat 2 entitas luar yaitu user sebagai pengguna sistem yang bisa melakukan proses pencarian otomatis. dan admin sebagai pengelola sistem, pada user terdapat beberapa alur data seperti data hasil cari, dan data keyword. Pada admin juga ada alur data seperti data login, data admin, dan Data Obat balita:



Gambar 3.2 Diagram Konteks Sitem Pencarian Obat

b. Data Flow Diagram level 1

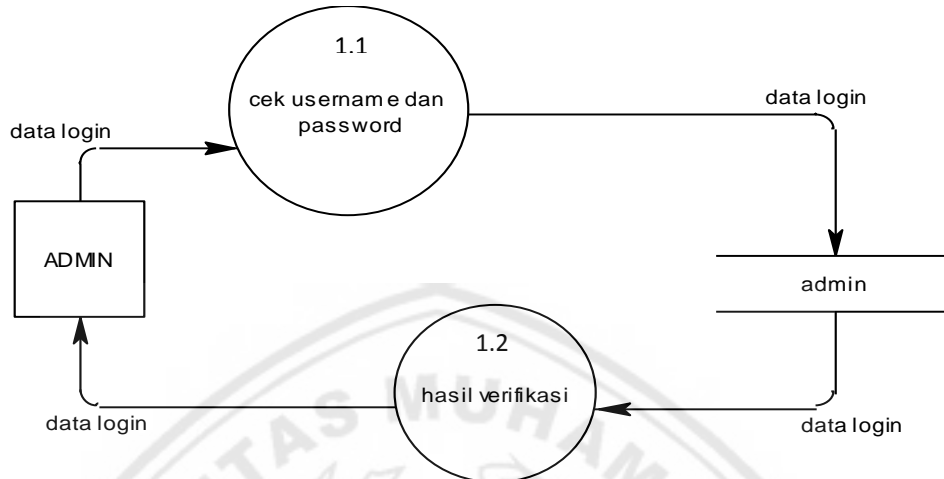
Pada Data Flow Diagram level 1 merupakan representasi dari data Diagram Konteks level 0 yang sudah dipartisi agar memberikan penjelasan yang lebih lengkap. Pada aplikasi pencarian otomatis ini terdapat empat proses seperti proses login, proses olah data, Data admin, olah data Obat balita, dan proses cosine similarity gambar 3.2.



Gambar 3.3 DFD Level 1

c. Data Flow Diagram Level 2 - Login

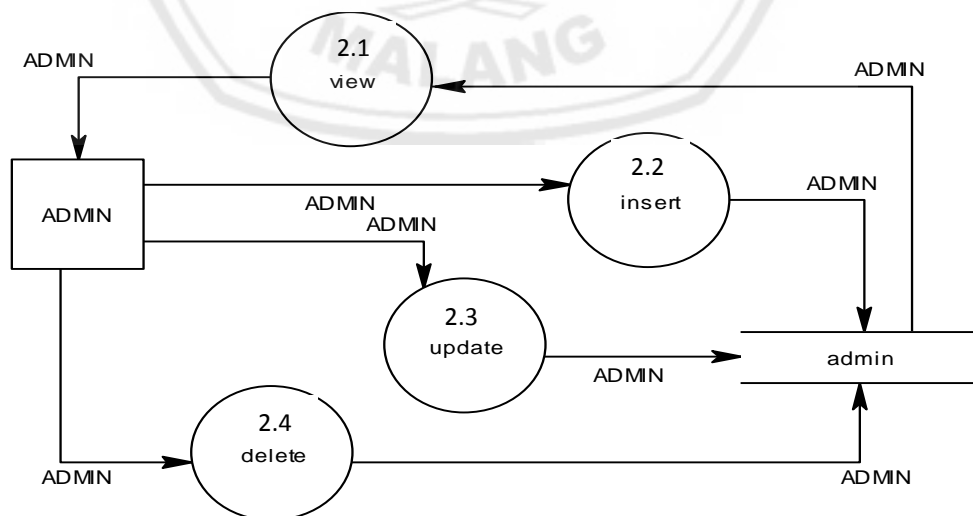
DFD level 2 Login ini merupakan penjabaran dari DFD level 1 yang memuat proses-proses yang ada dalam sistem. Yaitu meliputi kelola cek username dan password, dan hasil verifikasi username dan password.



Gambar 3.4. DFD Level 2 Proses Login

d. Data Flow Diagram Level 2 - Manajemen Admin

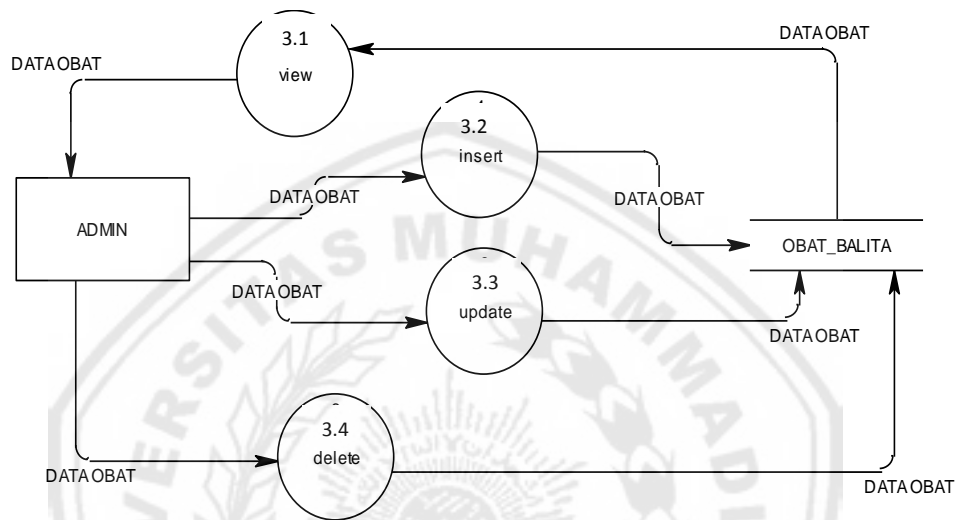
Pada Data Flow Diagram level 2 proses olah data admin atau manajemen data admin ini penjabarannya lebih detail dari proses olah data admin. Untuk proses ini ada empat proses meliputi proses tambah data admin, proses ubah admin, proses hapus admin, dan proses view data admin. Pada proses ini ada satu database yaitu database data admin, yang bisa dilihat pada gambar berikut:



Gambar 3.5. DFD Level 2 - Manajemen Admin

e. Data Flow Diagram Level 2 - Manajemen Obat balita

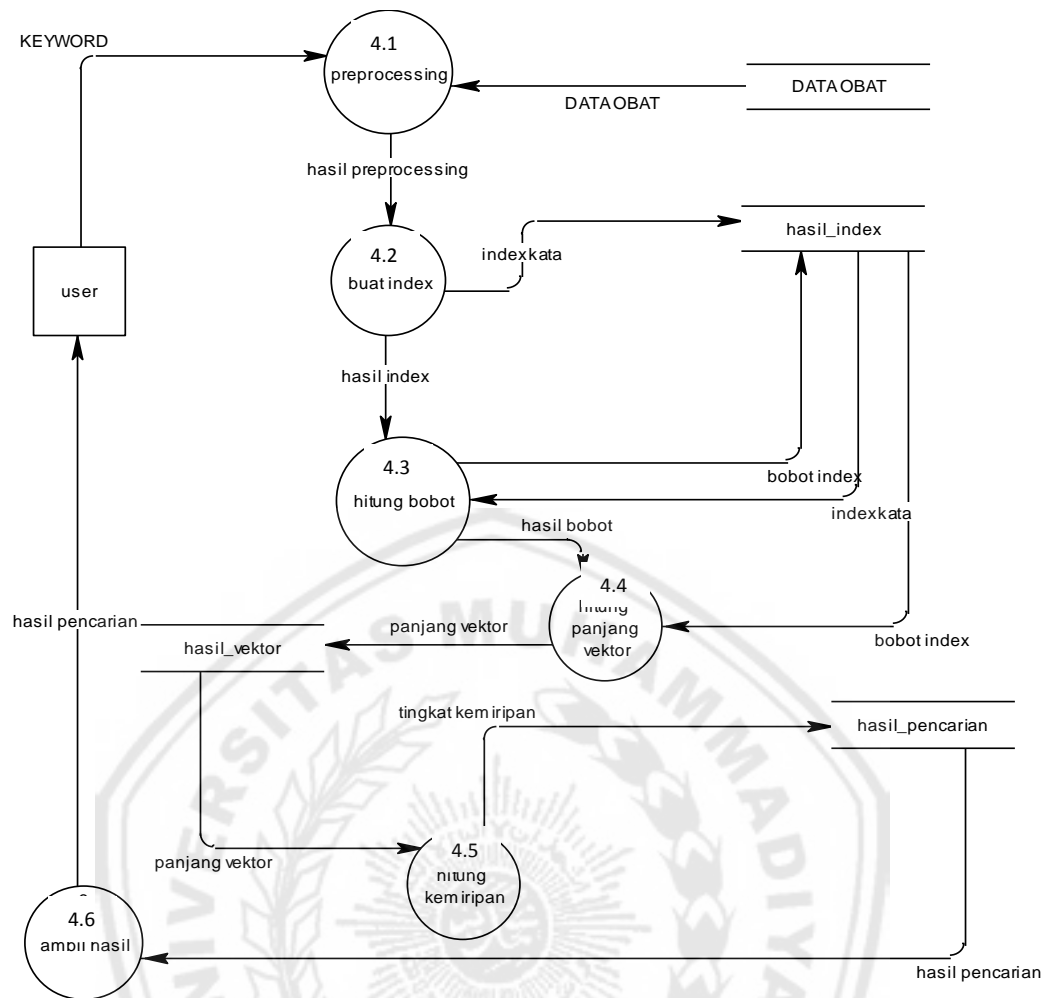
Pada Data Flow Diagram level 2 proses olah data Data obat menerangkan secara terperinci dan lebih detail dari proses olah data Data obat. Pada proses ini terdapat empat proses meliputi proses tambah data data obat, proses ubah data obat, proses hapus data obat, dan proses view data obat. Dalam proses ini terdapat satu database yaitu database data obat, yang bisa dilihat pada gambar 3.5



Gambar 3.6 DFD Level 2 - Manajemen Obat balita.

f. Data Flow Diagram Level 2 - Cosine Similarity

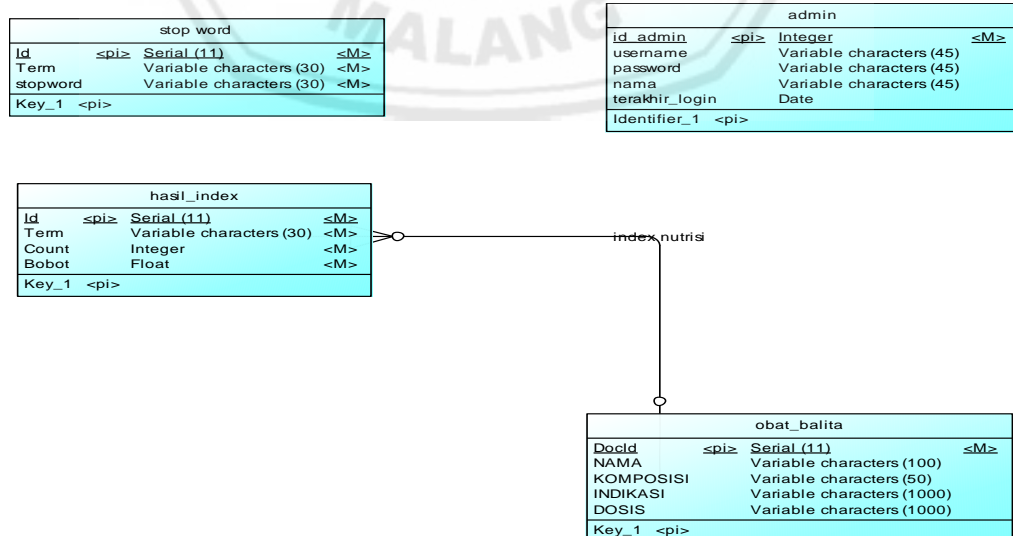
Pada hasil turunan dari diagram level 1 cosine similarity terdapat enam proses yaitu proses preprocessing yang berfungsi untuk melakukan filtering, dan stemming pada awal proses inputan pertanyaan yang dilakukan oleh user, proses buat index yang berfungsi untuk mengindex semua data obat dan pertanyaan menjadi perkata, proses hitung bobot dengan menggunakan metode tf-idf pada tiap index yang telah tersimpan pada table hasil index, proses hitung panjang vector yang hasilnya akan disimpan pada table hasil vector, proses hitung kemiripan dengan menggunakan rumus cosine similarity yang hasilnya akan disimpan pada tabel hasil dan proses view hasil yang berfungsi untuk menampilkan data hasil pencarian kepada user dengan data yang telah di urutkan sebelumnya.



Gambar 3.7. DFD Level 2 – Cosine Similarity.

3.3 Desain Database

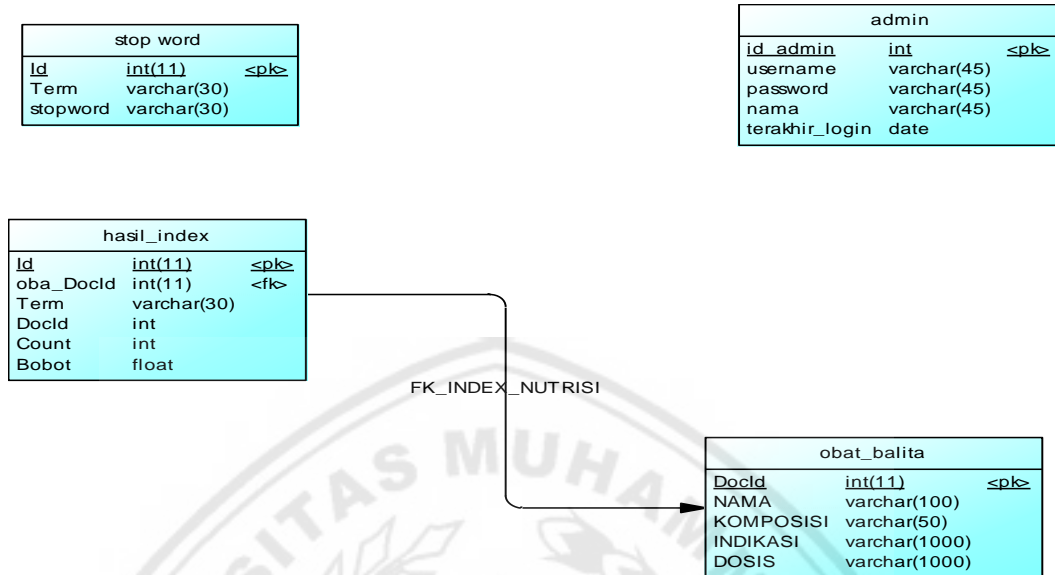
3.3.1 Diagram Konseptual (Conceptual Data Model)



Gambar 3.8 Diagram Konseptual Database

Pada gambar 3.8 diagram konseptual database memiliki 4 entitas yaitu stop word, admin, hasil index dan obat balita.

3.3.2 Diagram Fisik Database



Gambar 3.9 Diagram Fisik Database

Pada gambar 3.9 diagram fisik database memiliki 4 entitas yaitu stop word, admin, hasil index dan obat balita.

3.4 Keterangan Tabel

3.4.1 Tabel Obat balita

Nama Tabel : Obat balita
 Primary Key : id_Obat balita
 Foreign Key : username
 Fungsi : Menyimpan data Obat balita

Tabel 3.9 Keterangan Tabel Obat balita

No	Nama Fild	Tipe Data	Keterangan
1	Id_Obat	Varchar	Primary Key
2	Nama_Obat	Varchar	Foreign Key
3	Komposisi	Varchar	Not Null
4	Indikasi	Varchar	Not Null
5	Dosis	Varchar	Not Null

3.4.2 Tabel Stopword

Nama Tabel : Stopword

Primary Key : id

Fungsi : Menyimpan data stopwords

Tabel 3.10 Keterangan Tabel stopwords

No	Nama Fild	Tipe Data	Keterangan
1	Id	Int	Primary Key
2	term	Varchar	Not Null
3	stopword	Varchar	Not Null

3.4.3 Tabel Admin

Nama Tabel : Admin

Primary Key : username

Fungsi : Menyimpan data admin

Tabel 3.11 Keterangan Tabel admin

No	Nama Fild	Tipe Data	Keterangan
1	Id_admin	Varchar	Primary Key
2	username	Varchar	Foreign Key
3	password	Varchar	Not Null
4	nama	Varchar	Not Null
5	Terakhir_login	Number	Not Null

3.4.4 Tabel Hasil Index

Nama Tabel : Hasil Index

Primary Key : id

Fungsi : Menyimpan hasil index

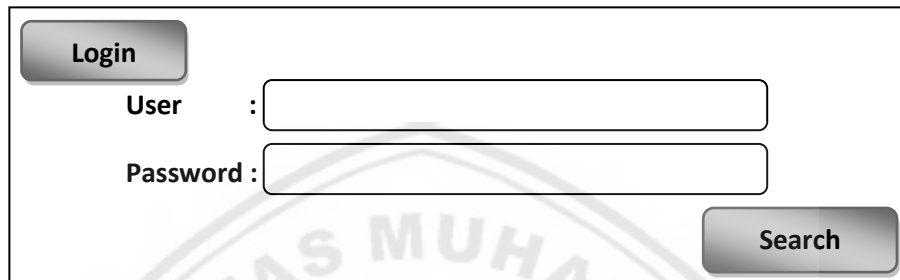
Tabel 3.12 Tabel Hasil Index

No	Nama Fild	Tipe Data	Keterangan
1	Id	Int	Primary Key
2	Id_Obat	Int	Foreign Key
3	term	Varchar	Not Null
4	docid	Int	Not Null

5	count	Int	Not Null
6	bobot	Float	Not Null

3.5 Desain Interface

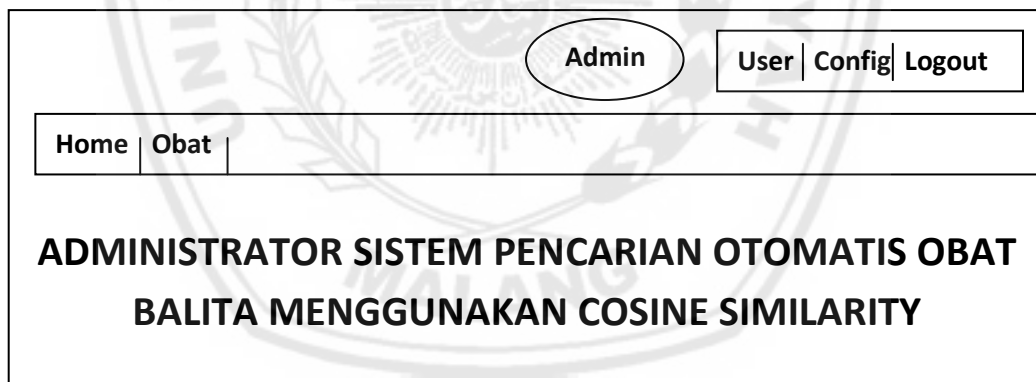
Suatu aplikasi harus mempunyai desain tampilan yang menarik, interaktif dan mudah dimengerti oleh user. Sehingga perlu didesain secara detail dan seksama. Berikut sebagai tampilan awal dari *interface admin*.



The login form consists of a rectangular container. Inside, there is a 'Login' button at the top left. Below it, there are two input fields: one for 'User' and one for 'Password'. To the right of the 'Password' field is a 'Search' button. The background of the form has a faint watermark of a university seal.

Gambar 3.10 Tampilan Login form

Pada gambar 3.10 merupakan tampilan yang pertama kali muncul ketika admin akan melakukan proses pengolahan data. Tampilan ini memiliki Login setelah itu admin memasukkan username dan password.



The admin dashboard features a header with a circular 'Admin' button and a navigation bar with 'User', 'Config', and 'Logout' links. Below the navigation bar is a secondary menu with 'Home' and 'Obat' links. The main content area displays the title 'ADMINISTRATOR SISTEM PENCARIAN OTOMATIS OBAT BALITA MENGGUNAKAN COSINE SIMILARITY' in bold, uppercase letters. A large, faint watermark of a university seal is visible in the background.

Gambar 3.11 Halaman Admin

Pada gambar 3.11 merupakan tampilan dari Halaman Admin yang akan menjelaskan tentang menu apa saja yang terdapat pada admin yaitu menu home, obat dan case folding.

No.	Nama Obat	Indikasi	Kontra Indikasi	Keterangan Lain	
					<input type="button" value="Edit"/> <input type="button" value="Del"/>
					<input type="button" value="Edit"/> <input type="button" value="Del"/>

Gambar 3.12 View dan Manajemen data Obat balita

Pada gambar 3.12 merupakan tampilan menu data obat yang berfungsi untuk mengelola data-data obat balita yang akan disimpan dalam database. Pada menu ini terdapat menu tambah data , menu edit dan menu delete.

Gambar 3.13 Desain Interface System User

Pada gambar 3.13 merupakan tampilan desain interface sytem user yang berfungsi untuk user melihat tampilan pencarian obat dan hasil pencarian obat.